

REMARKS

Claims 1-29 are pending in the present application. Claims 9, 17, 20, and 23 are amended. Reconsideration of the claims is respectfully requested.

Amendments were made to the specification to correct errors and to clarify the specification. No new matter has been added by any of the amendments to the specification.

Also, applicants have submitted proposed corrections to drawings labeled **Figures 1 and 2**, as suggested by the Examiner in red ink. These changes will be incorporated into a formal set of drawings upon approval of the proposed changes by the examiner.

I. Drawing Objections

The Office Action objects to the drawings for failing to comply with 37 CFR 1.84(p)(5). The drawings and specification are amended to overcome these objections. The Office Action states that reference number 30 of Figure 1 is not mentioned in the description. Applicant notes that reference number 30 appears on page 12, line 16.

II. 35 U.S.C. § 102, Anticipation

The Office Action rejects claims 1-3, 5, 6, 13, and 17 under 35 U.S.C. § 102 as being anticipated by Weikart et al. (EP 0831409 A2), hereinafter referred to as "*Weikart*." This rejection is respectfully traversed.

With respect to claims 1, 13, and 17, the Office Action states:

As per claim 1, 13, and 17, Weikart teaches a method and computer program product for extending the capabilities of a web server, comprising the steps of:

- sending a request from a client to the web server, the request including an address for the code module needed to service the request (see abstract, lines 4-6, pg. 2, lines 36-40, pg. 3, lines 34-56 and see Fig. 2);
- if the code module is unavailable at the web server, having the web server use the address to request the code module from the publishing server (see pg. 4, lines 5-7);
- installing the code module at the server; and performing the request at the web server using the install code module (see pg. 4, lines 20-33).
- serving a response to the request back to the client (see pg. 3, lines 41-45).

Office Action, dated September 16, 2002. Applicant respectfully disagrees. The prior art describes the well known process of installing a plug-in at a client. More specifically, *Weikart* teaches a computerized method for browser-based electronic messaging. A Web page includes an embedded tag specifying a messaging extension program. When the Web page is requested from a browser program of a client, the extension program is retrieved. The extension program is identified in the tag using a Uniform Resource Locator (URL). The present invention, however, teaches a hitherto unknown process of using information in a client request (previously unknown to the server) to retrieve and install code **at the server**.

As such, *Weikart* does not teach or suggest "sending a request from a client to the web server, the request including an address for a code module needed to service the request," as recited in representative claim 1. The cited portion of *Weikart* states:

The web page 200 includes an embedded HTML tag 210. The tag 210 specifies parameters for a "browser extension." A browser extension is a program which extends the capabilities of the browser application program 101. The specified extension can dynamically be retrieved by the client computer 110 via the network 150 and stored in the memory of the client. The extension is executed from within the application environment of the browser. However, the extension cannot run as stand-alone application software.

Weikart, page 3, lines 46-50. Thus, it is the web page 200 that includes the address of the extension program, not the request as in the present invention.

Furthermore, *Weikart* fails to teach or suggest "installing the code module at the web server," as recited in representative claim 1. The cited portion of *Weikart* states retrieving and executing or installing the code module on the client, rather than on the server as in representative claim 1. The applied prior art does not teach or suggest each and every claim limitation; therefore, claims 1, 13, and 17 are not anticipated by *Weikart*.

Since claims 2, 3, 5, and 6 depend from claim 1, the same distinctions between *Weikart* and the invention recited in claim 1 apply for these claims. Therefore, the rejection of claims 1-3, 5, 6, 13, and 17 under 35 U.S.C. § 102 is overcome.

Furthermore, *Weikart* does not teach, suggest, or give any incentive to make the needed changes to reach the presently claimed invention. *Weikart* actually teaches away

from the presently claimed invention because it teaches identifying an extension program in a Web page and installing the extension program on the client, as opposed to identifying a code module in a request and installing the code module on a server, as in the presently claimed invention. Absent the Office Action pointing out some teaching or incentive to implement a method or program product for extending the capabilities of a Web server, one of ordinary skill in the art would not be led to modify *Weikart* to reach the present invention when the reference is examined as a whole. Absent some teaching, suggestion, or incentive to modify *Weikart* in this manner, the presently claimed invention can be reached only through an improper use of hindsight using the applicants' disclosure as a template to make the necessary changes to reach the claimed invention.

The Office Action rejects claim 9 under 35 U.S.C. § 102 as being anticipated by Aggarwal et al. (US Patent No. 5,924,116), hereinafter referred to as "*Aggarwal*." This rejection is respectfully traversed.

With respect to claim 9, the Office Action states:

As per claim 9, Aggarwal discloses a servlet that allows a client program to upload additional program code and execute the code within the server (col. 5, lines 13-22). Therefore, Aggarwal implicitly discloses during a given Web transaction, uploading a code module from the client to the web server; and at the web server, using the uploaded code module as needed to service a given request from the web client.

Office Action, dated September 16, 2002. Applicant agrees that *Aggarwal* generally teaches a servlet that allows a client program to upload additional program code and execute the code within the server. However, *Aggarwal* does not teach the specific steps of claim 9. More particularly, claim 9, as amended, recites:

9. A method for enabling a web client to add functionality to a web server on an as-needed basis, comprising the steps of:

receiving a request from a client, **the request identifying a code module required to process the request;**

responsive to a determination that the code module is not available at the web server, uploading a code module from the client to the web server; and

at the web server, using the uploaded code module as needed to service a given request from the web client. [emphasis added]

Aggarwal does not teach or suggest a request identifying a code module required to process the request, uploading a code module from the client to the web server **responsive to a determination that the code module is not available at the web server**, and using the uploaded code module at the web server as needed to service a given request from the web client. The applied prior art does not teach or suggest each and every limitation of claim 9, particularly as amended; therefore claim 9 is not anticipated by *Aggarwal*.

III. 35 U.S.C. § 103, Obviousness

The Office Action rejects claims 14, 16, and 18 under 35 U.S.C. § 103 as being unpatentable over *Weikart* in view of Lee et al. (US Patent No. 6,167,522), hereinafter referred to as “*Lee*.” This rejection is respectfully traversed.

Claims 14 and 16 are allowable at least by virtue of their dependence on claim 13. Claim 16 is also dependent upon claim 15, which is not rejected as being unpatentable over *Weikart* alone or in combination with *Lee*. Therefore, the rejection of claim 16 is improper and should be withdrawn. Claim 18 is allowable at least by virtue of its dependence on claim 17. As stated above with respect to claims 13 and 17, *Weikart* fails to teach or suggest receiving a request from a client to the web server, the request including an address for a code module needed to service the request and installing the code module at the web server.

Lee teaches a method and apparatus for providing security for servers executing application programs received via a network. *Lee* also fails to teach or suggest receiving a request from a client to the web server, the request including an address for a code module needed to service the request; therefore, *Lee* does not make up for the deficiencies of *Weikart*. Thus, it follows that *Weikart* and *Lee*, taken alone or in combination, fail to teach or suggest the present invention as further limited in claims 14, 16, and 18. While *Weikart* and *Lee* individually teach some of the claim limitations, neither reference teaches a request including an address for a code module needed to service the request.

Even assuming, *arguendo*, that *Weikart* and *Lee* could be combined, the combination would not result in the claimed invention. Instead, a combination of *Weikart*

and *Lee* would result in a method for installing a plug-in at the client from a server, wherein the server also authenticates application programs received via a network.

Therefore, the rejection of claims 14, 16, and 18 under 35 U.S.C. § 103 is overcome.

The Office Action rejects claims 4, 7, 8, 15, 19, 23, and 25-28 under 35 U.S.C. § 103 as being unpatentable over *Weikart* in view of Gosling et al. (EP 0810524 A1), hereinafter referred to as "*Gosling*." This rejection is respectfully traversed.

Claims 23 and 28 recite subject matter addressed above with respect to claims 1, 13, and 17 and are allowable for the same reasons. Additionally, claims 23 and 28 recite other combinations of features not suggested by the prior art. Claims 4, 7, 8, 15, 19, 25, and 26 are allowable at least by virtue of their dependence on claims 1, 13, 17, and 23. As stated above with respect to claims 13 and 17, *Weikart* fails to teach or suggest receiving a request from a client to the web server, the request including an address for a code module needed to service the request and installing the code module at the web server.

Gosling teaches an apparatus and method for processing servlets. *Gosling* also fails to teach or suggest receiving a request from a client to the web server, the request including an address for a code module needed to service the request; therefore, *Gosling* does not make up for the deficiencies of *Weikart*. In both *Weikart* and *Gosling*, the Web page itself must identify the code module for servicing the request. In other words, in *Weikart* and *Gosling*, however, the code module is static and inflexible. In contradistinction, the present invention sends a request from a client to the server, wherein the request identifies the code module. Therefore, a request received from an in-vehicle device may identify a different code module from a request received from a personal digital assistant.

As such, *Weikart* and *Gosling*, taken alone or in combination, fail to teach or suggest each and every claim limitation; therefore, the claimed invention, particularly as further limited in claims 4, 7, 8, 15, 19, 23, and 25-28, is not rendered obvious by *Weikart* and *Gosling*. While *Weikart* and *Gosling* individually teach some of the claim

limitations, neither reference teaches a request including an address for a code module needed to service the request.

Even assuming, *arguendo*, that *Weikart* and *Gosling* could be combined, the combination would not result in the claimed invention. Instead, a combination of *Weikart* and *Gosling* would result in a method for installing a plug-in at the client from a server, wherein the server also receives servlets from other servers in a network. However, there is no suggestion to combine *Weikart* and *Gosling* to install code at a server, wherein the code is identified in a request from a client.

The Office Action rejects claims 10, 11, and 29 under 35 U.S.C. § 103 as being unpatentable over *Aggarwal* in view of *Weikart*. This rejection is respectfully traversed.

Claims 10 and 11 are allowable at least by virtue of their dependence on claim 9. Claim 29 is allowable at least by virtue of its dependence on claim 28, which is not rejected as being unpatentable over *Aggarwal* alone or in combination with *Weikart*. Therefore, the rejection of claim 29 is improper and should be withdrawn. As stated above with respect to claim 9, *Aggarwal* fails to teach or suggest a request identifying a code module required to process the request, uploading a code module from the client to the web server responsive to a determination that the code module is not available at the web server, and using the uploaded code module at the web server as needed to service a given request from the web client.

Weikart teaches a computerized method for browser-based electronic messaging. *Weikart* also fails to teach or suggest receiving a request from a client, the request identifying a code module required to process the request; therefore, *Weikart* does not make up for the deficiencies of *Aggarwal*. Thus, it follows that *Aggarwal* and *Weikart*, taken alone or in combination, fail to teach or suggest the present invention as further limited in claims 10 and 11.

More particularly, with respect to claim 11, the Office Action states:

As per claim 11, although *Aggarwal* shows substantial features of the claimed invention (discussed above), it fails to explicitly disclose:

- wherein the code module translates data into a given proprietary format and serves the translated data back to the pervasive computing client.

Weikart discloses a MIME “type” of plug-in extension required to

process page 200 requested by the client (see pg. 3, lines 41-43 and lines 56-57, pg. 4, lines 1-7 and lines 20-33). Therefore, Weikart implicitly discloses wherein the code module translates data into a given proprietary format and serves the translated data back to the pervasive computing client.

Therefore, one of ordinary skill in the art at the time the invention was made would have found it obvious to implement or incorporate a plug-in extension in Aggarwal's system enabling a client/user to received data in the desired format in a timely and efficient manner.

Office Action, dated September 16, 2002. Applicant respectfully disagrees. While *Weikart* may disclose a MIME type of plug-in extension, *Weikart* does not teach or suggest "wherein the code module translates data into a given proprietary format and **serves the translated data back to the pervasive computing client,**" as recited in claim 11. The applied prior art, taken alone or in combination, fails to teach the claim limitations; therefore claim 11 is not rendered obvious by *Aggarwal* and *Weikart*.

Therefore, the rejection of claims 10, 11, and 29 under 35 U.S.C. § 103 is overcome.

The Office Action rejects claim 12 under 35 U.S.C. § 103 as being unpatentable over *Aggarwal* in view of *Gosling*. This rejection is respectfully traversed.

With respect to claim 12, the Office Action states:

As per claim 12, although *Aggarwal* shows substantial features of the claimed invention (discussed above), it fails to disclose:

- wherein the code module conforms to a given application programming interface (API)

However, *Gosling* discloses a web server that interacts with servlets through an application-programming interface (API) (see pg. 3, lines 13-17 and pg. 5, lines 18-42). Therefore, *Gosling* implicitly discloses wherein the code module conforms to a given application programming interface (API).

Therefore, one of ordinary skill in the art at the time the invention was made would have found it obvious to implement or incorporate a web server that interacts with servlets through an API in *Weikart*'s method providing an adequate means for dynamically generating information.

Office Action, dated September 16, 2002. Applicant respectfully disagrees. Claim 12 is allowable at least by virtue of its dependence on claim 9. As stated above with respect to claim 9, *Aggarwal* fails to teach or suggest a request identifying a code module required

to process the request, uploading a code module from the client to the web server responsive to a determination that the code module is not available at the web server, and using the uploaded code module at the web server as needed to service a given request from the web client.

Gosling teaches an apparatus and method for processing servlets. *Gosling* also fails to teach or suggest receiving a request from a client to the web server, the request including an address for a code module needed to service the request; therefore, *Gosling* does not make up for the deficiencies of *Aggarwal*. As such, *Aggarwal* and *Gosling*, taken alone or in combination, fail to teach or suggest each and every claim limitation; therefore, the claimed invention, particularly as further limited in claim 12, is not rendered obvious by *Aggarwal* and *Gosling*.

Furthermore, the Office Action proposes implementing or incorporating a Web server that interacts with servlets through an API in *Weikart*'s method. It is unclear whether *Weikart* is intended to be included in the rejection. However, Applicant assumes that claim 9 is rejected as being obvious over *Aggarwal* in view of *Gosling*, as in the statement of the rejection, and that "*Weikart*" in the body of the rejection should be "*Aggarwal*."

Incidentally, the Office Action also states:

As per claim 16, although *Weikart* shows substantial features of the claimed invention (discussed above), it fails to disclose:

- wherein the step of authenticating includes applying a given key to information retrieved from the publishing server.

However, in an analogous art, Lee discloses:

- wherein the step of authenticating includes applying a given key to information retrieved from the publishing server (col. 4, lines 14-36). Therefore, Lee implicitly discloses wherein the step of authenticating includes applying a given key to information retrieved from the publishing server.

Therefore, one of ordinary skill in the art would have found it obvious to implement and incorporate a verification procedure in *Weikart*'s method enabling a server to have access privileges to resources in order to process a request of a client/user.

Office Action, dated September 16, 2002. It is unclear how the Office Action intends to reject claim 16, since the only reference mentioned is *Weikart* and there is no statement of

the rejection for this claim. Furthermore, claim 16 is previously rejected as being obvious over *Weikart* and *Lee*. However, claim 16 is allowable at least by virtue of its dependence on claims 13-15. As stated above with respect to claim 13, *Weikart* fails to teach or suggest receiving a request from a client to the web server, the request including an address for a code module needed to service the request and installing the code module at the web server. Thus, it follows that *Weikart* fails to teach or suggest the present invention as further limited in claim 16.

Therefore, the rejection of claim 16 under 35 U.S.C. § 103 is overcome.

The Office Action rejects claim 20 under 35 U.S.C. § 103 as being unpatentable over *Weikart* in view of *Aggarwal*. This rejection is respectfully traversed.

Weikart teaches a computerized method for browser-based electronic messaging. A Web page includes an embedded tag specifying a messaging extension program. When the Web page is requested from a browser program of a client, the extension program is retrieved. The extension program is identified in the tag using a Uniform Resource Locator (URL). However, *Weikart* fails to teach or suggest “means for receiving a request from a client, the request identifying a code module required to process the request,” as recited in amended claim 20.

Aggarwal teaches a servlet that allows a client program to upload additional program code and execute the code within the server. However, *Aggarwal* also does not teach or suggest receiving a request from a client, the request identifying a code module required to process the request; therefore, *Aggarwal* does not make up for the deficiencies of *Weikart*. As such, *Weikart* and *Aggarwal*, taken alone or in combination, fail to teach or suggest each and every claim limitation; therefore, the claimed invention is not rendered obvious by *Weikart* and *Aggarwal*.

The Office Action rejects claim 21 under 35 U.S.C. § 103 as being unpatentable over *Weikart* in view of *Aggarwal* and further in view of *Lee*. This rejection is respectfully traversed.

Claim 21 is allowable at least by virtue of its dependence on claim 20. As stated above with respect to claim 20, *Weikart* and *Aggarwal*, taken alone or in combination, fail to teach or suggest each and every claim limitation. *Lee* teaches a method and

apparatus for providing security for servers executing application programs received via a network. *Lee* also fails to teach or suggest receiving a request from a client, the request identifying a code module required to process the request; therefore, *Lee* does not make up for the deficiencies of *Weikart* and *Aggarwal*. Thus, it follows that *Weikart*, *Aggarwal*, and *Lee*, taken alone or in combination, fail to teach or suggest the present invention as further limited in claim 21.

Therefore, the rejection of claim 21 under 35 U.S.C. § 103 is overcome.

The Office Action rejects claim 22 under 35 U.S.C. § 103 as being unpatentable over *Weikart* in view of *Aggarwal* and further in view of *Gosling*. This rejection is respectfully traversed.

Claim 22 is allowable at least by virtue of its dependence on claim 20. As stated above with respect to claim 20, *Weikart* and *Aggarwal*, taken alone or in combination, fail to teach or suggest each and every claim limitation. *Gosling* teaches an apparatus and method for processing servlets. *Gosling* also fails to teach or suggest receiving a request from a client, the request identifying a code module required to process the request; therefore, *Gosling* does not make up for the deficiencies of *Weikart* and *Aggarwal*. As such, *Weikart*, *Aggarwal*, and *Gosling*, taken alone or in combination, fail to teach or suggest each and every claim limitation; therefore, the claimed invention, particularly as further limited in claim 22, is not rendered obvious by *Weikart*, *Aggarwal*, and *Gosling*.

Therefore, the rejection of claim 22 under 35 U.S.C. § 103 is overcome.

The Office Action rejects claim 24 under 35 U.S.C. § 103 as being unpatentable over *Weikart* in view of *Gosling* and further in view of *Lee*. This rejection is respectfully traversed.

Claim 24 is allowable at least by virtue of its dependence on claim 23. As stated above with respect to claim 23, *Weikart* and *Gosling*, taken alone or in combination, fail to teach or suggest each and every claim limitation; therefore, the claimed invention as further recited in claim 23 is not rendered obvious by *Weikart* and *Gosling*.

Lee teaches a method and apparatus for providing security for servers executing application programs received via a network. *Lee* also fails to teach or suggest receiving a request from a client, the request identifying a code module required to process the

request; therefore, *Lee* does not make up for the deficiencies of *Weikart* and *Gosling*. Thus, it follows that *Weikart*, *Gosling*, and *Lee*, taken alone or in combination, fail to teach or suggest the present invention as further limited in claim 24.

Therefore, the rejection of claim 24 under 35 U.S.C. § 103 is overcome.

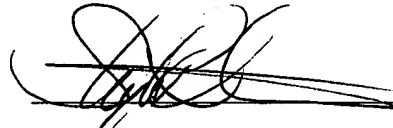
VII. Conclusion

It is respectfully urged that the subject application is patentable over the prior art of record and is now in condition for allowance.

The Examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the Examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: Dec. 16, 2002

Respectfully submitted,



Stephen R. Tkacs
Reg. No. 46,430
Carstens, Yee & Cahoon, LLP
P.O. Box 802334
Dallas, TX 75380
(972) 367-2001
Agent for Applicants